REMARKS

Claims 1-20 were pending at the time of examination. Claims 1, 3, 8, 10, 15 and 17 have been amended. Claims 2, 9 and 16 have been canceled. No new matter has been added. The Applicant respectfully requests reconsideration based on the foregoing amendments and these remarks.

Claim Rejections – 35 U.S.C. § 101

Claims 15-20 were rejected under 35 U.S.C § 101 as being directed to non-statutory matter, by being directed towards software, per se. The Applicant has amended claim 15 to include a computer, and respectfully submits that claim 15 as amended is directed to statutory matter. Claims 16-20 depend directly or indirectly from claim 15, and are thus directed to statutory matter for at least he same reasons as claim 15. Thus it is respectfully submitted that claims 15-20 as amended are directed to statutory subject matter and it is requested that the rejection under 35 U.S.C § 101 be withdrawn.

Claim Rejections – 35 U.S.C. § 103

Claims 1-20 were rejected under 35 U.S.C § 103(a) as being unpatentable over U.S. Patent No. 5,724,570 to Zeller et al (hereinafter "Zeller"), in view of SQL-92 Specification section 6.10 (hereinafter "SQL-92"). The Applicant respectfully traverses these rejections.

As was mentioned in the previous Office Action Response, in general, the Applicant's invention, as defined in claim 1, is directed to caching a datatype of an SQL template. This allows users to quickly determine the datatype of a SQL expression and perform datatype resolution in constant time, even for a complex tree of SQL expressions. Zeller, on the other hand, is directed to completely eliminating SQL subqueries from relational database queries by transforming the subqueries into semantically equivalent forms. Zeller achieves this by processing a tree of SQL expressions in a "normalizer" that applies a number of "normalization rules" in order to create a "normalized and syntactically transformed but equivalent tree," which is subsequently optimized and stored on a secondary storage memory hard drive, where the normalized tree can be accessed, when needed.

Thus, while both the Applicant's invention and Zeller are directed to SQL expression resolution, the Applicant's invention only concerns itself with resolving a datatype of an SQL expression, whereas the Zeller is directed to completely eliminating SQL queries. The Applicant has amended claim 1 to more clearly define the Applicant's invention. Claim 1, as amended, recites:

> "A method for datatype caching of an SQL template including two or more tokens, each token referencing a different SQL template or an object, comprising:
>
> replacing each of the two or more tokens of the SQL template with an associated cast function to form a converted SQL template;
>
> acquiring a datatype of the converted SQL template; and
>
> storing the acquired datatype of the converted SQL template with the SQL template."

Claim 1, as amended, clearly specifies that the SQL template must have two or more tokens, and that each of these tokens references another SQL template or an object. Claim 1 also includes the limitations of claim 2, which is now canceled, and thus specifies that the tokens are replaced with a cast function to form a converted SQL template. Respectfully, Zeller does not show any datatype caching, and does not show any replacing of tokens with cast functions. Instead, Zeller shows a set of normalization rules that are used for completely eliminating subqueries, none of which uses a cast function to replace tokens in an SQL expression.

The Examiner contends that the "acquiring a datatype..." limitation is shown in Zeller, since "the datatype is known as the query is acted on, such as BOOLEAN" (Office Action, page 6), and refers to Zeller col. 8, lines 27-31. However, the cited passage merely describes one of Zeller's normalization rules, and states that an <u>assumption of the rule</u> is that "the subquery has a BOOLEAN result." That is, this particular normalization rules simply assumes that a result of a query will be BOOLEAN. Merely making such an assumption in a normalization rule does not anticipate or render obvious determining a datatype of a SQL template. Furthermore, the Applicant respectfully disagrees with the Examiner's assertion that the datatype is known as the query is acted on. Whereas this, in theory, might be the case for a SQL template that only has a single reference or token, it is not the case for SQL templates that include two or more tokens, as is required by claim 1.

It is clear that the cast function in SQL-92 was well known at the time of the invention. The cast function is even discussed in paragraph [0044] of the Applicant's specification. The issue is rather whether the Examiner has shown that Zeller and SQL-92 are readily combinable by a person of ordinary skill in the art to yield a predicable result. The Applicant respectfully contends that such a showing has not been made. Zeller discloses a set of normalization functions to be used in the elimination of nested tree queries. There would be no reason to do an additional datatype determination in Zeller – in fact Zeller does not even mention any issues with datatype determination - or to look at any additional cast functions for such a purpose. Furthermore, as was mentioned in the previous response, once the tree in Zeller has been converted with the normalization functions, the tree (and thus the datatype of the SQL template) will be fully resolved and stored on a hard drive for future accesses. That is, there will be no need for a user to do a separate datatype determination using functions other than Zeller's normalization functions. There would also be no need to quickly access the datatype of an SQL template by accessing a datatype in a cache, since the datatype would be determined in Zeller for all normalized trees, as part of the application of the normalization rules. Using a cast function in combination with Zeller would merely add unnecessary complexity to the techniques described in Zeller, with no added resulting benefit.

Finally, the combination of the references must also teach or suggest all the claim limitations. Even if it were possible to combine Zeller and SQL-92, the combination still would not teach the limitations of claim 1. The preamble and the first limitation have been discussed above. The second clause of claim 1 specifies "acquiring a datatype of the converted SQL template." Zeller, respectfully, does not disclose <u>acquiring</u> a datatype of the <u>converted</u> SQL template (i.e., a template with two or more cast functions). At best, the cited section of Zeller specifies that a normalization rule (NR3) can be used on the condition that a subquery has a Boolean result. This subquery is not equivalent to the converted SQL template recited in claim 1. The last limitation of claim 1, "storing the datatype of the converted SQL template with the SQL template" is also absent in Zeller. What is stored in Zeller is an optimized version of the normalized tree, in which subqueries have been eliminated. No mention is made of storing a datatype for a converted SQL template. For at least these reasons, the rejection of claim 1 is unsupported by the art and should be withdrawn.

Claims 3-7 depend directly or indirectly from claim 1, and are thus not anticipated or rendered obvious for at least the reasons discussed above. It is respectfully requested that the rejection under 35 U.S.C § 103(a) be withdrawn for these claims.

For reasons substantially similar to those set forth above, the Applicant respectfully contends that the rejection of the computer program product claims 8 and 10-14 and the system claims 15 and 17-20 is unsupported by the cited art and should be withdrawn.

Conclusion

The Applicant believes that all pending claims are allowable and respectfully requests a Notice of Allowance for this application from the Examiner. Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,
MOLLBORN PATENTS

/Fredrik Mollborn/

Fredrik Mollborn
Reg. No. 48,587

2840 Colby Drive
Boulder, CO 80305
(303) 459-4527